



PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

Docket No: A8494

CHEUNG, Tom Thuan

Appln. No.: 09/364,370

Group Art Unit: 2126

Confirmation No.: 9357

Examiner: The T. HO

Filed: July 30, 1999

For: AN OBJECT IN, OBJECT OUT TECHNIQUE

**SUBMISSION OF APPELLANT'S BRIEF ON APPEAL**

**MAIL STOP APPEAL BRIEF - PATENTS**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**RECEIVED**

JUN 27 2003

Technology Center 2100

Sir:

Submitted herewith please find an original and two copies of Appellant's Brief on Appeal. The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account. A duplicate copy of this paper is attached.

Respectfully submitted,

SUGHRUE MION, PLLC  
Telephone: (202) 293-7060  
Facsimile: (202) 293-7860

WASHINGTON OFFICE



23373

PATENT TRADEMARK OFFICE

Date: June 25, 2003

for Billy Carter Raulin REG. NO. 52,156  
J. Warren Lytle, Jr.  
Registration No. 39,283



PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

Docket No: A8494

CHEUNG, Tom Thuan

Appln. No.: 09/364,370

Group Art Unit: 2126

Confirmation No.: 9357

Examiner: The T. HO

Filed: July 30, 1999

RECEIVED

JUN 27 2003

For: AN OBJECT IN, OBJECT OUT TECHNIQUE

Technology Center 2100

APPELLANTS' BRIEF ON APPEAL UNDER 37 C.F.R. § 1.192

MAIL STOP APPEAL BRIEF - PATENTS

Commissioner for Patents

P.O. Box 1450

Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 1.192, Appellant submits the following:

I. REAL PARTY IN INTEREST

The real party in interest in this appeal is International Business Machines Corporation ("IBM") of Armonk, New York, the assignee.

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellant, Appellant's legal representative, or the assignee that will directly affect or be directly affected by, or have a bearing on, the Board's decision in this appeal.

06/26/2003 JADD01 00000060 194880 09364370

01 FC:1402 320.00 DA

### III. STATUS OF CLAIMS

Claims 1 - 30 are the claims pending in the application and are the subject of this appeal.

A copy of the claims on appeal are set forth in an attached Appendix.

### IV. STATUS OF AMENDMENTS

All Amendments submitted have been entered. A Response Under 37 C.F.R. § 1.116 was filed on March 14, 2002, in response to a Final Office Action (Paper No. 10), and the Examiner indicates in the Advisory Action (Paper No. 14) that the Response was considered.

### V. SUMMARY OF THE INVENTION

The invention relates to object oriented methods, apparatuses, and articles of manufacture for processing input objects to generate output objects.

In many software development environments, one or more development teams work on components of a system that must interface with systems developed by other development teams. In conventional environments each team must understand the logic of the systems that depend on the components being developed by those teams. For example, a team responsible for maintaining a banking system that accesses a database must be aware of the developments being made by the team responsible for the database. In these conventional environments someone from each development team must understand the internal logic of functions within the other system to ensure that the components interface properly. *See* page 1. Accordingly, there is a need to process an input object, from one of the teams, to generate an output object that is

APPELLANTS' BRIEF ON APPEAL  
UNDER 37 C.F.R. § 1.192  
U.S. Appln. No.: 09/364,370

compatible with the other development team's system to eliminate the need for knowledge in each team of the internal workings of the other team's system.

The present invention solves that problem by using an input object that contains both data and a function that is executable on a computer. A controller object receives the input object, determines the object's type, and based on the object type ascertains whether the object satisfies requirements for that type of object. If it does satisfy those predetermined requirements, the function contained within the input object is executed to produce an output object.

An example use of the techniques described in the application is set forth at page 7 of the specification. In this example one team of developers is developing a product that works with a server that is developed by another team. Here, the team developing the product defines an input object that can be used by the server team. The product team creates functions that are contained in the input object and defines the data that must be input to those functions. The server team uses the input object to supply information to the product team. The product team receives the input object, executes the internal functions to generate data compatible with the server team's server system.

Referring to Fig. 3, an example of the input object 300 is received by a controller object 302 that uses the data and functions within the input object to generate an output object 304. The output object provides the input information in a form that is usable with a system being developed by a team other than the team using the input object. In this manner, the internal workings of separate systems that must interface with each other need not be mastered by other development teams.

VI. ISSUES

- 1) Whether claims 1–8, 11–18, and 21–28 are unpatentable under 35 U.S.C. § 103(a) over Dean (U.S. Patent No. 6,003,094) in view of Smithies (U.S. Patent No. 5,544,255).
- 2) Whether claims 9, 19 and 29 are unpatentable under 35 U.S.C. § 103(a) over Dean in view of Smithies and Aditham (U.S. Patent No. 6,378,00).
- 3) Whether claims 10, 20 and 30 are unpatentable under 35 U.S.C. § 103(a) over Dean in view of Smithies and Nakai (U.S. Patent No. 6,253,248).

VII. GROUPING OF CLAIMS

The claims do not stand or fall together and arguments for patentability of each group of claims, identified below, are set forth in this brief.

Group I: Claims 1–7, 11–17, and 21–27, each of which stand or fall together.

Group II: Claims 8–10, 18–20 and 28–30, each of which stand or fall together.

VIII. ARGUMENTS

Appellant submits the following arguments in support of patentability.

A. Issue 1, Group I: The Rejection of the Claims of Group I Under § 103(a) over Dean in view of Smithies:

The claims in Group I, rejected under 35 U.S.C. § 103(a) over Dean in view of Smithies, recite limitations not found in either of those reference, whether considered alone or in combination. Accordingly, the claims in Group I are not rendered unpatentable and the Board should reverse the rejection.

APPELLANTS' BRIEF ON APPEAL  
UNDER 37 C.F.R. § 1.192  
U.S. Appln. No.: 09/364,370

*1. The Dean Reference Relates To An Airline Ticket Booking System That Transmits Messages Containing Only Data, Between Object Oriented Programs*

The Dean reference describes techniques for a client workstation, running front-end booking software, to complete transactions on a transaction processing system, such as an airline ticket booking system. Dean, in Fig. 1, shows a client computer 10 with a JAVA enabled network browser 11, running a client object that sends messages to a gateway workstation 14. The gateway workstation relays data in those messages to a transaction server 12, such as a CICS transaction processing system.

Dean, in Fig. 3, shows a logical flow diagram that illustrates a method for a client workstation to operate with the generic JAVA gateway to book a ticket. *See also*, col. 3, line 60 through col. 5, line 9. The left-hand portion of Fig. 3 shows steps taken by the client workstation, the middle portion shows steps taken by the JAVA gateway, and the right-hand portion shows steps taken by the CICS transaction processing system. Dean describes, beginning at col.3, line 64, the client workstation using two object-oriented objects, “encodefixedformat” and “writemessage” to create a message that is sent to the JAVA gateway. *See* col. 4, lines 27-28.

The message generated by the client workstation contains only data and Dean does not disclose or suggest that it includes any functions that are executable by a computer. The “encodefixedformat.” object writes a fixed part of the message that classifies the type of message. For example, it writes the value “BookTktRelay” in the fixed part of the message. *See* col. 4, lines 19-23 and 11-13. The “writemessage” object writes the request-specific portion of the message. *See* col. 4, lines 24-26. As described in steps (7) and (8) beginning at col. 4, line

31, the gateway uses the fixed part of the message to determine the type of the gateway object to use to process that type of request. The "request-specific data" is then read from the message in step (9), and the request is processed by the CICS transaction processing system.

Nowhere does Dean teach or suggest that the message, illustrated by the arrow in Fig. 3 extending from the client workstation to the gateway, contains anything but data.

*2. The Smithies Reference Relates To Capturing and Verifying A Handwritten Signature*

The Smithies reference is directed to a computer-based system for capturing and verifying a handwritten signature. Smithies discloses a signature envelope 10, shown in Fig. 1 that includes information that represents a person's signature. See col. 4, lines 7-32, 49-57 and col. 7, lines 35-67. A signature verification module 6 receives the signature envelope and verifies the signature using a plurality of templates stored in a template database. See col. 4, lines 64-col. 5, line 4. Each template includes act-of-signing statistics for a person and the known identity of that person.

*3. The Rejection of Claims 1-8, 11-18, and 21-28 under 35 U.S.C. § 103(a) over Dean in view of Smithies*

In the final Office Action, the Examiner rejects claims 1-8, 1-8 and 21-28 (Group I) as being unpatentable over Dean in view of Smithies. The Examiner asserts that Dean teaches all the limitations of claim 11, for example, except for determining if an input object satisfies a predetermined requirement. The Examiner relies on Smithies for teaching that limitation.

APPELLANTS' BRIEF ON APPEAL  
UNDER 37 C.F.R. § 1.192  
U.S. Appln. No.: 09/364,370

Neither Dean nor Smithies, either alone or in combination, teaches or suggests “receiving an input object, wherein the received input object contains input data and one input function executable on a computer” as required by claims 1, 11 and 21. The Examiner asserts that Dean teaches that the gateway workstation 4 receives “an input object (client-side request object, lines 1–2 column 4) from the client workstation (10, Fig. 1)”. The Examiner refers to the objects “Myflight.encodefixedformat” and “Myflight.writemessage” shown in Fig. 3 as functions contained within the “input object” that the gateway receives from the client workstation.

Contrary to the Examiner’s assertion, the only thing Dean discloses that the gateway receives from the client workstation is a message. *See* col. 4, line 29 (“The generic Java Gateway receives the message.”) And that message only contains data, namely, the fixed format and the request-specific parts of the message. *See* col. 4, lines 7-26. Dean does not teach or suggest that the message contains a function that can be executed on a computer. Contrary to the Examiner’s assertion, the message does not contain the objects “Myflight.encodefixedformat” and “Myflight.writemessage” shown in Fig. 3. Rather, those objects correspond to the “encodefixedformat” and “writemessage” objects that reside in the client workstation. *Id.*

Accordingly, Dean does not teach or suggest receiving an input object that contains input data and one input function that is executable on a computer, since the Dean message contains only data and does not contain a function executable on a computer.

The Examiner relies on Smithies for disclosing verification features. However, even when considered in combination with Smithies, a Dean/Smithies combination does not teach or



APPELLANTS' BRIEF ON APPEAL  
UNDER 37 C.F.R. § 1.192  
U.S. Appln. No.: 09/364,370

suggest receiving an input object that contains data and one input function, as required by the claims.

For at least these reasons Appellant respectfully requests the Board to reverse the Examiner's rejections of the claims in Group I.

B. Issue 1, Group II: The Rejection of the Claims 8, 18 and 28 Under § 103(a) over Dean in view of Smithies:

Claims 8, 18 and 28 (in Group II) are also rejected under 35 U.S.C. § 103(a) over Dean in view of Smithies, but contain an additional limitation not present in the claims in Group I, namely that of regulating a flow of received input objects.

In the final Office Action the Examiner asserts that Dean discloses regulating the flow of objects by referring to Dean's steps 5 and 6 shown in Fig. 3 and discussed at col. 4, lines 4-29. However, as discussed above, those portions of Dean merely describe the generation and transmission of a data message. Although Dean's uses the word "flow" (*see* col. 4, line 4) Dean does not teach or suggest "regulating a flow of received input objects" as required by the claims in Group II. Rather Dean's use of the word "flow" appears to be used to describe passing a message to the CICS transaction processing system so that it flows through the JAVA gateway.

It is respectfully submitted that Dean and Smithies, alone or in combination, do not teach all the limitations of the claims in Group II. Accordingly, the Board should reverse the Examiner's rejection of claims 8, 18 and 28.

APPELLANTS' BRIEF ON APPEAL  
UNDER 37 C.F.R. § 1.192  
U.S. Appln. No.: 09/364,370

C. Issue 2, Group II: The Rejection of the Claims 9, 19 and 29 Under § 103(a) over Dean in view of Smithies and Aditham:

Claims 9, 19 and 29 (Group II) are rejected under 35 U.S.C. § 103(a) over Dean in view of Smithies and further in view of Aditham (U.S. Patent No. 6,378,001). It is respectfully submitted that these references, alone or in combination, do not teach all the limitations of claims 9, 19 and 29.

Claims 9, 19 and 29 incorporate by reference all the limitations of claim 8 and further recite “wherein regulating the flow comprises storing some of the received input objects in a queue.”

Aditham relates to a system for enabling collaboration between application programs. The reference describes a collaborative session that is represented by a session object which receives all messages generated by the programs and transmits the messages to all programs participating in the session. *See Summary.*

The Examiner relies on Aditham for storing received objects in a FIFO queue. Col. 6, lines 19-20. However, it is respectfully submitted that Aditham does not disclose regulating a flow of received input objects as required by claims 8, 18 and 28. Accordingly, even if the teachings of Dean and Smithies were combined with Aditham as the Examiner asserts, all the limitations of the claims in Group II would not be met. For at least these reasons Appellant respectfully requests the Board to reverse the rejections of the claims 9, 19 and 29.

D. Issue 3, Group II: The Rejection of the Claims 10, 20 and 30 Under § 103(a) over Dean in view of Smithies and Nakai:

APPELLANTS' BRIEF ON APPEAL  
UNDER 37 C.F.R. § 1.192  
U.S. Appln. No.: 09/364,370

Claims 10, 20 and 30 (Group II) are rejected under 35 U.S.C. § 103(a) over Dean in view of Smithies and further in view of Nakai (U.S. Patent No. 6,253,248). It is respectfully submitted that these references, alone or in combination, do not teach all the limitations of claims 10, 20 and 30.

Claims 10, 20 and 30 include the limitation of "returning some of the received input objects to a sender; and requesting that the sender resend the received input objects at a later time."

Nakai relates to a proxy server that performs protocol conversion and determines the sever that contains the information that will satisfy a request from a client. *See* Summary.

The Examiner relies on Nakai for disclosing that a sender can be requested to resend an input object at a later time. Col. 13, lines 40-41. That portion of Nakai discloses the proxy server authenticating a user's identity by checking the user name and password. If the authentication fails, such as when the supplied user name and password do not match that user's authorization information, "the proxy server requests the client 107 of the client machine A 1101 to resend the request." *See* col. 13, lines 32-44.

It is respectfully submitted that Nakai does not disclose regulating a flow of received input objects as required by claims 8, 18 and 28, and incorporated by reference in claims 10, 20 and 30. Accordingly, even if the teachings of Dean and Smithies were combined with Nakai as the Examiner asserts, all the limitations of the claims in Group II would not be met. For at least these reasons Appellant respectfully requests the Board to reverse the rejections of claims 10, 20 and 30.

APPELLANTS' BRIEF ON APPEAL  
UNDER 37 C.F.R. § 1.192  
U.S. Appln. No.: 09/364,370

The present Brief on Appeal is being filed in triplicate. Unless a check is submitted herewith for the fee required under 37 C.F.R. §1.192(a) and 1.17(c), please charge said fee to Deposit Account No. 19-4880.

The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account.

Respectfully submitted,

*Billy Carter Rauler*  
for J. Warren Lytle, Jr. *Reg. No. 52,156*  
Registration No. 39,283

SUGHRUE MION, PLLC  
Telephone: (202) 293-7060  
Facsimile: (202) 293-7860

WASHINGTON OFFICE



23373

PATENT TRADEMARK OFFICE

Date: June 25, 2003

## APPENDIX

### CLAIMS 1-30 ON APPEAL:

1. (Amended) A method of producing an output object, the method comprising the steps of:

receiving an input object, wherein the received input object contains input data and one input function executable on a computer;

determining a type of the received input object;

based on the determined type, ascertaining whether the received input object satisfies one or more predefined requirements; and

when it is ascertained that the received input object satisfies each predefined requirement, executing the input function on a computer.

2. The method of claim 1, wherein the step of ascertaining further comprises ascertaining whether the received input object satisfies one or more predefined requirements by executing one or more verification functions.

3. The method of claim 2, wherein a source code for each verification function is located in a predefined section of a controller object source code.

4. The method of claim 1, further comprising the step of producing an output object by using a result produced by the executed input function.

5. The method of claim 4, wherein the received input object is received from an application, and wherein the method further comprises the step of returning the output object to the application.

6. The method of claim 4, wherein the received input object is received from a user and wherein the method further comprises the step of returning the output object to the user.

7. The method of claim 1, wherein the step of receiving comprises receiving a plurality of input objects, wherein each received input object contains an input function, and wherein each input function has a predefined signature.

8. The method of claim 7, wherein the method further comprises the step of regulating a flow of received input objects.

9. The method of claim 8, regulating the flow comprises storing some of the received input objects in a queue.

10. The method of claim 8, wherein regulating the flow comprises the steps of: returning some of the received input objects to a sender; and requesting that the sender re-send the received input objects at a later time.

11. (Amended) An apparatus for producing an output object, comprising:

a computer;

one or more computer programs, performed by the computer, for receiving an input object, wherein the received input object contains input data and one input function executable on a computer, determining a type of the received input object, based on the determined type, ascertaining whether the received input object satisfies one or more predefined requirements, and when it is ascertained that the received input object satisfies each predefined requirement, executing the input function on a computer.

12. The apparatus of claim 11, wherein the means of ascertaining further comprises ascertaining whether the received input object satisfies one or more predefined requirements by executing one or more verification functions.

13. The apparatus of claim 12, wherein a source code for each verification function is located in a predefined section of a controller object source code.

14. (Amended) The apparatus of claim 11, wherein the apparatus further comprises one or more computer programs executing on the computer for producing an output object by using a result produced by the executed input function.

15. The apparatus of claim 14, wherein the received input object is received from an application, and wherein the apparatus further comprises one or more computer programs, performed by the computer for returning the output object to the application.

16. The apparatus of claim 14, wherein the received input object is received from a user, and wherein the apparatus further comprises one or more computer programs, performed by the computer for returning the output object to the user.

17. The apparatus of claim 11, wherein the means of receiving comprises receiving a plurality of input objects, wherein each received input object contains an input function, and wherein each input function has a predefined signature.

18. The apparatus of claim 17, wherein the apparatus further comprises one or more computer programs, performed by the computer for regulating a flow of received input objects.

19. The apparatus of claim 18, wherein regulating the flow comprises storing some of the received input objects in a queue.

20. The apparatus of claim 18, wherein the flow comprises:  
one or more computer programs, performed by the computer for returning some of the received input objects to a sender, and requesting that the sender re-send the received input objects at a later time.



21. (Amended) An article of manufacture comprising a computer program carrier readable by a computer and embodying one or more instructions executable by the computer to perform method steps for producing an output object, the method comprising the steps of:

receiving an input object, wherein the received input object contains input data and one input function executable on a computer;

determining a type of the received input object;

based on the determined type, ascertaining whether the received input object satisfies one or more predefined requirements; and

when it is ascertained that the received input object satisfies each predefined requirement, executing the input function on a computer.

22. The article of manufacture of claim 21, wherein the step of ascertaining further comprises ascertaining whether the received input object satisfies one or more predefined requirements by executing one or more verification functions.

23. The article of manufacture of claim 22, wherein a source code for each verification function is located in a predefined section of a controller object source code.

24. The article of manufacture of claim 21, wherein the method further comprises the step of producing an output object by using a result produced by the executed input function.

25. The article of manufacture of claim 24, wherein the received input object is received from an application, and wherein the method further comprises the step of returning the output object to the application.

26. The article of manufacture of claim 24, wherein the received input object is received from a user, and wherein the method further comprises the step of returning the output object to the user.

27. The article of manufacture of claim 21, wherein the step of receiving comprises receiving a plurality of input objects, wherein each received input object contains an input function, and wherein each input function has a predefined signature.

28. The article of manufacture of claim 27, wherein the method further comprises the step of regulating a flow of received input objects.

29. The article of manufacture of claim 28, wherein regulating the flow comprises storing some of the received input objects in a queue.

30. The article of manufacture of claim 28, wherein regulating the flow comprises the steps of:

returning some of the received input objects to a sender; and

requesting that the send re-send the received input objects at a later time.